

0280

Docket No. 500.38506X00

#12

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant(s): WATANABE, ET AL.
Serial No.: 09/559,060
Filed: April 28, 2000
Title: COOPERATIVE SYSTEM AND METHOD THEREFOR
Group:

LETTER CLAIMING RIGHT OF PRIORITY

Honorable Commissioner of
Patents and Trademarks
Washington, D.C. 20231

May 11, 2000

Sir:

Under the provisions of 35 USC 119 and 37 CFR 1.55, the
applicant(s) hereby claim(s) the right of priority based on:

Japanese Patent Application No.(s) 11-123508
Filed: April 30, 1999

A certified copy of said Japanese Patent Application is
attached.

Respectfully submitted,

ANTONELLI, TERRY, STOUT & KRAUS, LLP



Carl I. Brundidge
Registration No. 29,621

CIB/ssr
Attachment

日 本 国 特 許 庁

PATENT OFFICE
JAPANESE GOVERNMENT

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office.

出 願 年 月 日
Date of Application:

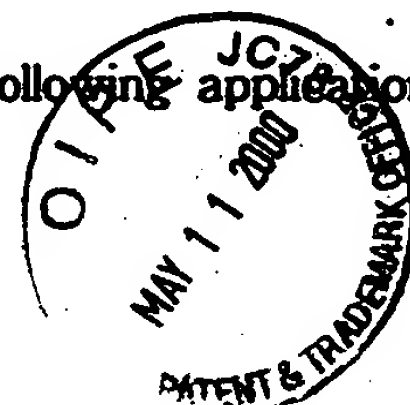
1999年 4月30日

出 願 番 号
Application Number:

平成11年特許願第123508号

出 願 人
Applicant(s):

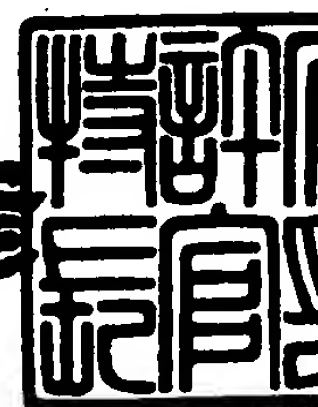
株式会社日立製作所



2000年 4月14日

特許庁長官
Commissioner,
Patent Office

近藤 隆彦



【書類名】 特許願

【整理番号】 DH410

【あて先】 特許庁長官殿

【国際特許分類】 G06F 15/00

【発明者】

 【住所又は居所】 神奈川県川崎市幸区鹿島田 8 9 0 番地 株式会社日立製作所 情報システム事業部内

 【氏名】 渡部 芳邦

【発明者】

 【住所又は居所】 神奈川県川崎市幸区鹿島田 8 9 0 番地 株式会社日立製作所 情報システム事業部内

 【氏名】 佐藤 信彦

【発明者】

 【住所又は居所】 神奈川県川崎市幸区鹿島田 8 9 0 番地 株式会社日立製作所 システム開発本部内

 【氏名】 北川 誠

【発明者】

 【住所又は居所】 神奈川県川崎市幸区鹿島田 8 9 0 番地 株式会社日立製作所 システム開発本部内

 【氏名】 橋本 哲也

【特許出願人】

 【識別番号】 000005108

 【氏名又は名称】 株式会社日立製作所

【代理人】

 【識別番号】 100096954

 【弁理士】

 【氏名又は名称】 矢島 保夫

【手数料の表示】

 【予納台帳番号】 022781

【納付金額】 21,000円

【提出物件の目録】

【物件名】	明細書	1
【物件名】	図面	1
【物件名】	要約書	1
【プルーフの要否】	要	

【書類名】 明細書

【発明の名称】 システム間連携システムおよび方法

【特許請求の範囲】

【請求項 1】

任意の数の第 1 の情報システムと任意の数の第 2 の情報システムとの間に介在して、該第 1 の情報システムと第 2 の情報システムとを連携させて動作させるシステム間連携システムであって、

第 1 の情報システムから送信されるメッセージを受け付ける受け付け手段と、

第 1 の情報システムからのメッセージで使用しているプロトコルと連携システム内で使用しているプロトコルとの間でプロトコル変換を行なうプロトコル変換手段と、

第 1 の情報システムからのメッセージを転送すべき送信先を取得する送信先取得手段と、

受け付けたメッセージを第 2 の情報システムに渡す経路として、特定プロトコル直結パスまたはそれ以外のパスの何れかを、受け付けたメッセージに基づいて選択するパス判定手段と、

前記パス判定手段で特定プロトコル直結パスが選択されたときは、受け付けた第 1 の情報システムからのメッセージに対し前記プロトコル変換手段によるプロトコル変換を施すことなく前記送信先に送信し、前記パス判定手段でそれ以外のパスが選択されたときは、受け付けた第 1 の情報システムからのメッセージに対し前記プロトコル変換手段によるプロトコル変換を施して前記送信先に送信する手段と

を備えたことを特徴とするシステム間連携システム。

【請求項 2】

任意の数の第 1 の情報システムと任意の数の第 2 の情報システムとの間に介在して、該第 1 の情報システムと第 2 の情報システムとを連携させて動作させるシステム間連携システムであって、

前記各第 1 の情報システムに対応する第 1 のアダプタと、前記各第 2 の情報システムに対応する第 2 のアダプタとを備え、

前記各第 1 のアダプタは、

第 1 の情報システムから送信されるメッセージを受け付ける受け付け手段と

第 1 の情報システムからのメッセージで使用しているプロトコルと連携システム内で使用しているプロトコルとの間でプロトコル変換を行なうプロトコル変換手段と、

第 1 の情報システムからのメッセージを転送すべき送信先を取得する送信先取得手段と、

受け付けたメッセージを第 2 の情報システムに渡す経路として、特定プロトコル直結パスまたはそれ以外のパスの何れかを、受け付けたメッセージに基づいて選択するパス判定手段と

を備え、

前記パス判定手段で特定プロトコル直結パスが選択されたときは、受け付けた第 1 の情報システムからのメッセージに対し前記プロトコル変換手段によるプロトコル変換を施すことなく前記送信先である第 2 のアダプタに送信し、前記第 1 のアダプタ内の前記パス判定手段でそれ以外のパスが選択されたときは、受け付けた第 1 の情報システムからのメッセージに対し前記プロトコル変換手段によるプロトコル変換を施して前記送信先に送信することを特徴とするシステム間連携システム。

【請求項 3】

任意の数の第 1 の情報システムと任意の数の第 2 の情報システムとの間に介在して、該第 1 の情報システムと第 2 の情報システムとを連携させて動作させるシステム間連携システムであって、

前記各第 1 の情報システムに対応する第 1 のアダプタと、与えられたメッセージにしたがって所定の複数の第 2 の情報システムにアクセスして一連の業務処理を実行させるフロー制御手段と、前記各第 2 の情報システムに対応する第 2 のアダプタとを備え、

前記各第 1 のアダプタは、

第 1 の情報システムから送信されるメッセージを受け付ける受け付け手段と

受け付けたメッセージを第 2 の情報システムに渡す経路として、特定プロトコル直結パスまたはそれ以外のパスの何れかを、受け付けたメッセージに基づいて選択するパス判定手段と、

第 1 の情報システムからのメッセージで使用しているプロトコルと連携システム内で使用しているプロトコルとの間でプロトコル変換を行なうプロトコル変換手段と、

第 1 の情報システムからのメッセージを転送すべき送信先として、前記特定プロトコル直結パスが選択されたときは前記第 2 のアダプタを送信先とし、それ以外のパスが選択されたときは前記第 2 のアダプタまたは前記フロー制御手段を送信先とする送信先取得手段と、

を備え、

前記パス判定手段で特定プロトコル直結パスが選択されたときは、受け付けた第 1 の情報システムからのメッセージに対し前記プロトコル変換手段によるプロトコル変換を施すことなく前記送信先である第 2 のアダプタに送信し、前記第 1 のアダプタ内の前記パス判定手段でそれ以外のパスが選択されたときは、受け付けた第 1 の情報システムからのメッセージに対し前記プロトコル変換手段によるプロトコル変換を施して前記送信先である第 2 のアダプタまたはフロー制御手段に送信することを特徴とするシステム間連携システム。

【請求項 4】

前記パス判定手段は、前記第 1 の情報システムと前記第 2 の情報システムとで同じプロトコルを使用しているとき、特定プロトコル直結パスを選択することを特徴とする請求項 1 から 3 の何れか 1 つに記載のシステム間連携システム。

【請求項 5】

前記特定プロトコル直結パス以外のパスが選択された場合、前記送信先取得手段は、前記受け付けたメッセージに対応して 1 つの第 2 の情報システムで処理するかまたは複数の第 2 の情報システムで処理するかを判定し、1 つの第 2 の情報システムで処理するときは送信先として当該第 2 の情報システムに対応する第 2 のアダプタを指定し、複数の第 2 の情報システムで処理するときは送信先として

前記フロー制御手段を指定することを特徴とする請求項 3 に記載のシステム間連携システム。

【請求項 6】

前記第 1 の情報システムにおけるメッセージ形式と前記第 2 の情報システムにおけるメッセージ形式との間のメッセージ変換を行なうメッセージ変換手段を、さらに備えた請求項 1 から 5 の何れか 1 つに記載のシステム間連携システム。

【請求項 7】

任意の数の第 1 の情報システムと任意の数の第 2 の情報システムとの間に介在して、該第 1 の情報システムと第 2 の情報システムとを連携させて動作させるシステム間連携方法であって、

第 1 の情報システムから送信されるメッセージを受け付けるステップと、

受け付けたメッセージを第 2 の情報システムに渡す経路として、特定プロトコル直結パスまたはそれ以外のパスの何れかを、受け付けたメッセージに基づいて選択するステップと、

特定プロトコル直結パスが選択されたときには、プロトコル変換を施すことなく、前記受け付けたメッセージを直接前記第 2 の情報システム側に転送するステップと、

特定プロトコル直結パス以外のパスが選択されたときには、前記第 1 の情報システムからのメッセージで使用しているプロトコルから連携システム内で使用しているプロトコルへとプロトコル変換を行なって転送するステップと

を備えたことを特徴とするシステム間連携方法。

【請求項 8】

任意の数の第 1 の情報システムと任意の数の第 2 の情報システムとの間に介在して、該第 1 の情報システムと第 2 の情報システムとを連携させて動作させるシステム間連携システムであって、前記各第 1 の情報システムに対応する第 1 のアダプタと、前記各第 2 の情報システムに対応する第 2 のアダプタとを備えた連携システムに適用する連携方法であって、

前記第 1 のアダプタにより、第 1 の情報システムから送信されるメッセージを受け付けるステップと、

受け付けたメッセージを第 2 の情報システムに渡す経路として、特定プロトコル直結パスまたはそれ以外のパスの何れかを、受け付けたメッセージに基づいて選択するステップと、

特定プロトコル直結パスが選択されたときは、受け付けた第 1 の情報システムからのメッセージに対しプロトコル変換を施すことなく送信先である第 2 のアダプタに送信するステップと、

特定プロトコル以外のパスが選択されたときは、受け付けた第 1 の情報システムからのメッセージに対し、第 1 の情報システムのプロトコルから連携システム内で使用しているプロトコルへとプロトコル変換を施し、該メッセージを転送すべき送信先を取得し、該メッセージを該送信先に転送するステップと

を備えたことを特徴とするシステム間連携方法。

【発明の詳細な説明】

【0 0 0 1】

【発明の属する技術分野】

本発明は、複数の情報システム間の機能連携を実現するシステム間連携システムおよび方法に関する。

【0 0 0 2】

【従来の技術】

従来より、各種の業務に対応した情報システムがそれぞれ個々に開発・運用されてきている。近年では、それら既存の情報システムを連携させることにより、さらに多種多様なサービスを実現する試みが為されている。

【0 0 0 3】

図 1 0 に、従来の情報システム間の連携の一例を示す。営業店システム 1 0 0 1 は営業店で各種の営業業務を行なう際に使用するシステム、勘定系システム 1 0 0 2 は銀行で金銭のやり取りに関するサービスを行なう際に使用するシステム、投資信託系システム 1 0 0 3 は証券会社で投資信託に関するサービスを行なう際に使用するシステムとする。例えば、営業店システム 1 0 0 1 で発生した金銭の授受に対し、営業店システム 1 0 0 1 から勘定系システム 1 0 0 2 にその情報を送信し、所定の口座間でその金銭授受を自動的に行なうためには、営業店シス

テム 1 0 0 1 と勘定系システム 1 0 0 2 とを接続し連携して動作させるようにする必要がある。また、インターネット・バンキング・システム 1 0 0 4 の導入により、各顧客がクライアント 1 0 0 5 からインターネット 1 0 0 6 を介してインターネット・バンキング・システム 1 0 0 4 に接続し各種の銀行サービスを受けるためには、インターネット・バンキング・システム 1 0 0 4 と銀行の勘定系システム 1 0 0 2 とを接続し連携して動作させるようにする必要がある。

【 0 0 0 4 】

以上のように各情報システムを接続して連携させる場合、従来は個別に対処していた。すなわち、連携させたい情報システムを個別に変更（機能追加など）して連携できるようにしていた。しかし、情報システムの種類は多種多様であり、それらの接続の組み合わせの数も非常に多い。したがって、連携させたいシステムを個別に変更する方式では、開発が面倒で、迅速・安価な多様化が困難である。また、図 1 0 の勘定系システム 1 0 0 2 のような他の複数のシステムと連携するシステムが変更された場合、他システムへの影響が大きく、システム間で整合性が取れなくなることも考えられる。

【 0 0 0 5 】

そこで、近年では各種の情報システムを経路制御やメッセージの変換を行なう機能をもつコアとなるシステムに接続し、このコアとなるシステムを介してシステム間を連携させる方式が提案されている。（本明細書では、このような方式をハブ・アンド・スポークと呼び、コアの部分をハブと呼ぶものとする。）

【 0 0 0 6 】

図 1 1 は、ハブ・アンド・スポークでの接続例を示す。ハブ 1 1 0 1 に、営業店システム 1 1 0 2、インターネット・バンキング・システム 1 1 0 3、コール・センタ・システム 1 1 0 4、投資信託系システム 1 1 0 5、CRM（カスタマ・リレーションシップ・マネジメント）システム 1 1 0 6、および勘定系システム 1 1 0 7 などが接続されている。営業店システム 1 1 0 2、インターネット・バンキング・システム 1 1 0 3、投資信託系システム 1 1 0 5、および勘定系システム 1 1 0 7 は、図 1 0 で説明した各システム 1 0 0 1 ～ 1 0 0 4 と同様のものである。コール・センタ・システム 1 1 0 4 は、例えばフリーダイヤルで顧客

からかかってきた電話をオペレータが受け、その顧客からの要求に応じて端末を操作して各種の業務を行なう、いわゆるコール・センタのシステムである。CRMシステム 1 1 0 6 は、顧客との関係を管理するシステムであり、例えば過去に顧客が購入した商品がどのようなものであるかを DB（データベース）に蓄積し、その購入状況に応じて最適な商品を提案するなど、顧客との一対一の適切な関係を築くための管理システムである。

【0 0 0 7】

これらの各システム 1 1 0 2 ~ 1 1 0 7 は、ハブ 1 1 0 1 に接続することにより、お互いのシステムを意識することなく連携できる。例えば、営業店システム 1 1 0 2 から他システムに業務を依頼するメッセージは、まずハブ 1 1 0 1 に入力する。ハブ 1 1 0 1 は、そのメッセージを送るべき相手先システムを判断し、その相手先システムに合わせたプロトコルおよびメッセージ形式に変換して、その相手先に送る。このように各システム間の差違をハブ 1 1 0 1 が吸収するので、各システムはハブ 1 1 0 1 に接続すれば連携が容易になる。新たなサービスを構築する場合、各システムは無変更（あるいはユーザインターフェースなどに関する少ない変更）で、ハブ内に各システムを連携させる処理手順を定義することにより、各システムの連携を容易に実現できる。

【0 0 0 8】

例えば、投資信託を個人が購入するときは、通常、自分の普通預金口座から金を引き出し（勘定系システムでの金の引き出し）、それを投資信託系に預け入れる（投資信託系システムに金を送って投資信託を購入）という操作が必要になるが、そのような複数のシステムにまたがった処理を簡単にハブ中に定義でき、これにより営業店やインターネット経由での複合サービスを実現できる。システムの構成や連携のパターンを変更する場合も、ハブ中の定義を変更することにより対処することができ、1つのシステムの変更が他システムに影響することがほとんどない。

【0 0 0 9】

【発明が解決しようとする課題】

ところで、上述したような従来のハブでは、どのようなメッセージを入力した

場合でも、ハブ内で経路判定しプロトコル変換およびメッセージ変換を施して相手先のシステムに転送する。したがって、メッセージ通信の時間や応答メッセージを得るまでのレスポンス時間が長くなるという問題があった。

【0 0 1 0】

本発明は、複数の情報システム間の機能連携を実現するハブ・アンド・スポークのシステムにおいて、ハブを経由するメッセージの通信時間や応答メッセージを得るまでのレスポンス時間を短くすることを目的とする。

【0 0 1 1】

【課題を解決するための手段】

上記目的を達成するため、本発明は、任意の数の第1の情報システムと任意の数の第2の情報システムとの間に介在して、該第1の情報システムと第2の情報システムとを連携させて動作させるシステム間連携システムであって、第1の情報システムから送信されるメッセージを受け付ける受け付け手段と、第1の情報システムからのメッセージで使用しているプロトコルと連携システム内で使用しているプロトコルとの間でプロトコル変換を行なうプロトコル変換手段と、第1の情報システムからのメッセージを転送すべき送信先を取得する送信先取得手段と、受け付けたメッセージを第2の情報システムに渡す経路として、特定プロトコル直結パスまたはそれ以外のパスの何れかを、受け付けたメッセージに基づいて選択するパス判定手段と、前記パス判定手段で特定プロトコル直結パスが選択されたときは、受け付けた第1の情報システムからのメッセージに対し前記プロトコル変換手段によるプロトコル変換を施すことなく前記送信先に送信し、前記パス判定手段でそれ以外のパスが選択されたときは、受け付けた第1の情報システムからのメッセージに対し前記プロトコル変換手段によるプロトコル変換を施して前記送信先に送信する手段とを備えたことを特徴とする。

【0 0 1 2】

また本発明は、任意の数の第1の情報システムと任意の数の第2の情報システムとの間に介在して、該第1の情報システムと第2の情報システムとを連携させて動作させるシステム間連携システムであって、前記各第1の情報システムに対応する第1のアダプタと、前記各第2の情報システムに対応する第2のアダプタ

とを備え、前記各第 1 のアダプタは、第 1 の情報システムから送信されるメッセージを受け付ける受け付け手段と、第 1 の情報システムからのメッセージで使用しているプロトコルと連携システム内で使用しているプロトコルとの間でプロトコル変換を行なうプロトコル変換手段と、第 1 の情報システムからのメッセージを転送すべき送信先を取得する送信先取得手段と、受け付けたメッセージを第 2 の情報システムに渡す経路として、特定プロトコル直結パスまたはそれ以外のパスの何れかを、受け付けたメッセージに基づいて選択するパス判定手段とを備え、前記パス判定手段で特定プロトコル直結パスが選択されたときは、受け付けた第 1 の情報システムからのメッセージに対し前記プロトコル変換手段によるプロトコル変換を施すことなく前記送信先である第 2 のアダプタに送信し、前記第 1 のアダプタ内の前記パス判定手段でそれ以外のパスが選択されたときは、受け付けた第 1 の情報システムからのメッセージに対し前記プロトコル変換手段によるプロトコル変換を施して前記送信先に送信することを特徴とする。

【0013】

また本発明は、任意の数の第 1 の情報システムと任意の数の第 2 の情報システムとの間に介在して、該第 1 の情報システムと第 2 の情報システムとを連携させて動作させるシステム間連携システムであって、前記各第 1 の情報システムに対応する第 1 のアダプタと、与えられたメッセージにしたがって所定の複数の第 2 の情報システムにアクセスして一連の業務処理を実行させるフロー制御手段と、前記各第 2 の情報システムに対応する第 2 のアダプタとを備え、前記各第 1 のアダプタは、第 1 の情報システムから送信されるメッセージを受け付ける受け付け手段と、受け付けたメッセージを第 2 の情報システムに渡す経路として、特定プロトコル直結パスまたはそれ以外のパスの何れかを、受け付けたメッセージに基づいて選択するパス判定手段と、第 1 の情報システムからのメッセージで使用しているプロトコルと連携システム内で使用しているプロトコルとの間でプロトコル変換を行なうプロトコル変換手段と、第 1 の情報システムからのメッセージを転送すべき送信先として、前記特定プロトコル直結パスが選択されたときは前記第 2 のアダプタを送信先とし、それ以外のパスが選択されたときは前記第 2 のアダプタまたは前記フロー制御手段を送信先とする送信先取得手段と、を備え、前

記パス判定手段で特定プロトコル直結パスが選択されたときは、受け付けた第 1 の情報システムからのメッセージに対し前記プロトコル変換手段によるプロトコル変換を施すことなく前記送信先である第 2 のアダプタに送信し、前記第 1 のアダプタ内の前記パス判定手段でそれ以外のパスが選択されたときは、受け付けた第 1 の情報システムからのメッセージに対し前記プロトコル変換手段によるプロトコル変換を施して前記送信先である第 2 のアダプタまたはフロー制御手段に送信することを特徴とする。

【 0 0 1 4 】

さらに本発明は、前記パス判定手段は、前記第 1 の情報システムと前記第 2 の情報システムとで同じプロトコルを使用しているとき、特定プロトコル直結パスを選択することを特徴とする。また、前記特定プロトコル直結パス以外のパスが選択された場合、前記送信先取得手段は、前記受け付けたメッセージに対応して 1 つの第 2 の情報システムで処理するかまたは複数の第 2 の情報システムで処理するかを判定し、 1 つの第 2 の情報システムで処理するときは送信先として当該第 2 の情報システムに対応する第 2 のアダプタを指定し、複数の第 2 の情報システムで処理するときは送信先として前記フロー制御手段を指定することを特徴とする。また、前記第 1 の情報システムにおけるメッセージ形式と前記第 2 の情報システムにおけるメッセージ形式との間のメッセージ変換を行なうメッセージ変換手段を、さらに備えたことを特徴とする。

【 0 0 1 5 】

本発明は、任意の数の第 1 の情報システムと任意の数の第 2 の情報システムとの間に介在して、該第 1 の情報システムと第 2 の情報システムとを連携させて動作させるシステム間連携方法であって、第 1 の情報システムから送信されるメッセージを受け付けるステップと、受け付けたメッセージを第 2 の情報システムに渡す経路として、特定プロトコル直結パスまたはそれ以外のパスの何れかを、受け付けたメッセージに基づいて選択するステップと、特定プロトコル直結パスが選択されたときには、プロトコル変換を施すことなく、前記受け付けたメッセージを直接前記第 2 の情報システム側に転送するステップと、特定プロトコル直結パス以外のパスが選択されたときには、前記第 1 の情報システムからのメッセー

ジで使用しているプロトコルから連携システム内で使用しているプロトコルへとプロトコル変換を行なって転送するステップとを備えたことを特徴とする。

【0 0 1 6】

また本発明は、任意の数の第 1 の情報システムと任意の数の第 2 の情報システムとの間に介在して、該第 1 の情報システムと第 2 の情報システムとを連携させて動作させるシステム間連携システムであって、前記各第 1 の情報システムに対応する第 1 のアダプタと、前記各第 2 の情報システムに対応する第 2 のアダプタとを備えた連携システムに適用する連携方法であって、前記第 1 のアダプタにより、第 1 の情報システムから送信されるメッセージを受け付けるステップと、受け付けたメッセージを第 2 の情報システムに渡す経路として、特定プロトコル直結パスまたはそれ以外のパスの何れかを、受け付けたメッセージに基づいて選択するステップと、特定プロトコル直結パスが選択されたときは、受け付けた第 1 の情報システムからのメッセージに対しプロトコル変換を施すことなく送信先である第 2 のアダプタに送信するステップと、特定プロトコル以外のパスが選択されたときは、受け付けた第 1 の情報システムからのメッセージに対し、第 1 の情報システムのプロトコルから連携システム内で使用しているプロトコルへとプロトコル変換を施し、該メッセージを転送すべき送信先を取得し、該メッセージを該送信先に転送するステップとを備えたことを特徴とする。

【0 0 1 7】

【発明の実施の形態】

以下、図面を用いて本発明の実施の形態を説明する。

【0 0 1 8】

図 1 は、本発明を適用した実施の形態であるハブ・アンド・スポーク・システムの概要を示す。クライアント側の情報システムとして、営業店システム 1 1 1、インターネット・バンキング・システム 1 1 2、およびコール・センタ・システム 1 1 3 が、ハブ 1 0 0 に接続されている。サーバ側の情報システムとして、勘定系システム 1 2 1、CRMシステム 1 2 2、および投資信託系システム 1 2 3 が、ハブ 1 0 0 に接続されている。これらの各システム 1 1 1～1 1 3、1 2 1～1 2 3 は、従来技術の欄で説明したのと同様の各業務を行なう情報システム

である。

【 0 0 1 9 】

ハブ 1 0 0 は、クライアント側アダプタ 1 0 1、サービス・ファインダ 1 0 2、フロー制御 A P 1 0 3、およびサーバ側アダプタ 1 0 4 を備える。これら各部は、分散オブジェクト環境を提供する C O R B A (Common Object Request Broker Architecture - C O R B A は Object Management Group が提唱する分散処理環境アーキテクチャの名称である) 仕様の通信制御手段 1 0 5 を介して基本的に相互にメッセージを交換する。特に、本システムでは、通信制御手段 1 0 5 を介さずにアダプタ 1 0 1 と 1 0 4 間でメッセージをやり取りするパスを備えていることを特徴とするが、それについては後述する。

【 0 0 2 0 】

クライアント側アダプタ 1 0 1 は、各クライアント側システムに対応して設けられ、クライアント側システムとの間のチャンネル I / F (インターフェース) 制御、各クライアント側システムのプロトコルとハブ内の C O R B A 仕様のプロトコルとの間のプロトコル変換、および各クライアント側システムのメッセージ形式とそのメッセージの送り先であるサーバ側システムのメッセージ形式との間のメッセージ変換などの機能を備える。

【 0 0 2 1 】

サーバ側アダプタ 1 0 4 は、各サーバ側システムに対応して設けられ、サーバ側システムとの間のサーバ I / F 制御、ラッパを利用したレガシー・システムへの接続、各サーバ側システムのプロトコルとハブ内の C O R B A 仕様のプロトコルとの間のプロトコル変換、および各サーバ側システムのメッセージ形式との間のメッセージ変換などの機能を備える。

【 0 0 2 2 】

サービス・ファインダ 1 0 2 は、メッセージ・ルーティング情報を管理する。例えば、クライアント側アダプタ 1 0 1 やフロー制御 A P 1 0 3 で受信したメッセージをどこに送信するか、その送信先を取得するときに、サービス・ファインダ 1 0 2 に問い合わせる。フロー制御 A P 1 0 3 は、ワーク管理を利用した複合サービスを提供する。すなわち、クライアント側アダプタ 1 0 1 で受信したメッ

セージに対応して、複数のサーバで連携した処理を行なう場合に、どのサーバ側アダプタ 1 0 4 にどのような順でメッセージを送信するか、その流れをフロー制御 A P 1 0 3 で制御する。

【0 0 2 3】

図 2 は、ハブ・アンド・スポーク・システムの実行環境の概要であり、図 1 のシステム概要のうちハブ 1 0 0 の構成をさらに詳細に記載したものである。図 2 では、ハブ 2 0 0 に、クライアント側システムとして営業店システム 2 0 3 とテラー端末 2 0 4 が接続され、サーバ側システムとして勘定系システム 2 0 5 と投資信託系システム 2 0 6 が接続されている。これらのシステム 2 0 3～2 0 6 は、図 1 で説明したクライアント側システムおよびサーバ側システムと同様のものである。テラー端末 2 0 4 は、いわゆる窓口端末であり、営業店窓口やコール・センタなどでオペレータが顧客の要求に応じて各種の情報を入力する端末である。ハブ 2 0 0 には、別拠点のハブ 2 0 1 や 2 0 2 が接続されている。ここではハブ 2 0 0 に接続されている情報システム間の連携について説明するが、ハブ 2 0 1, 2 0 2 に接続されている情報システムとの間の連携も同様に行なえる。マネージャ 2 1 0 は、ハブ 2 0 0 内の各部の運用管理、システム構成管理、およびログ取得指定制御などを行なう。

【0 0 2 4】

クライアント側システム 2 0 3, 2 0 4 から送信されたメッセージは、ハブ 2 0 0 を介してサーバ側システム 2 0 5, 2 0 6 に送信されるが、ハブ 2 0 0 内ではメッセージの経路（パス）として 3 つのパスを用意している。通常パス、アダプタ直結パス、および特定プロトコル直結パスの 3 つである。

【0 0 2 5】

通常パスについて説明する。アダプタ 2 3 0 は、テラー端末 2 0 4 に対応したアダプタである。テラー端末 2 0 4 から送信されたメッセージは、アダプタ 2 3 0 で受信される。アダプタ 2 3 0 は、テラー端末 2 0 4 からのメッセージを受信すると、3 つのパスのうちどのパスを利用するかを当該メッセージに応じて決定する。ここでは、通常パスを利用するメッセージであったとする。その場合、アダプタ 2 3 0 は、プロトコル変換機能 2 3 1 により当該メッセージのプロトコル

をハブ内部のプロトコルに変換し、送信先取得（宛先問い合わせ）機能 2 3 2 によりサービス・ファインダ 2 5 0 にそのメッセージをどこに送信するか送信先を問い合わせる。ファインダ 2 5 0 は、ハブ 2 0 0 内のアダプタ 2 2 0, 2 3 0, 2 7 0, 2 8 0 およびフロー制御 A P 2 6 0 の構成情報やメッセージ送付先システムの管理情報などを管理している。通常パスの場合、ファインダ 2 5 0 は、メッセージをフロー制御 A P 2 6 0 に送信することを指示する。アダプタ 2 3 0 は、それを受けて、必要に応じてメッセージ変換機能 2 3 3 によりメッセージ変換を行ない、ハブ内メッセージ送受信機能 2 3 4 により当該メッセージをフロー制御 A P 2 6 0 に送信する。メッセージ変換を行なう場合は、共通サービス 2 4 0 のメッセージ変換エンジン 2 4 1 やコード変換エンジン 2 4 2 を利用する。また、ハブ 2 0 0 内での処理については、ログ取得機能 2 4 3 により、処理履歴がログとして記録される。

【 0 0 2 6 】

フロー制御 A P 2 6 0 は、受け取ったメッセージに応じて、幾つかのサーバ側システムを決められた流れにしたがって順にアクセスし、当該メッセージに応じた処理を行なう。どのサーバ側システムをアクセスするかは、ファインダ 2 5 0 に問い合わせその管理情報を利用して決定する。ここでは、まず勘定系システム 2 0 5 にアクセスして所定の処理を行ない（2 6 1）、次に投資信託系システム 2 0 6 にアクセスして所定の処理を行なう（2 6 2）ものとする。勘定系システム 2 0 5 へのアクセス（2 6 1）では、勘定系システム 2 0 5 に対応するアダプタ 2 7 0 に所定のメッセージを送り、それに対する応答メッセージを得る。その後、投資信託系システム 2 0 6 へのアクセス（2 6 2）では、投資信託系システム 2 0 6 に対応するアダプタ 2 8 0 に所定のメッセージを送り、それに対する応答メッセージを得る。得られた応答メッセージは、必要に応じて加工し、フロー制御 A P 2 6 0 から元のアダプタ 2 3 0 に返す。アダプタ 2 3 0 は、プロトコル変換やメッセージ変換などの必要な処理を行なって、対応するテラー端末 2 0 4 に応答メッセージを返す。

【 0 0 2 7 】

勘定系システム 2 0 5 へのアクセス処理（2 6 1）によりフロー制御 A P 2 6

0から送信されたメッセージは、勘定系システム205に対応するアダプタ270のハブ内メッセージ送受信機能274により受信される。アダプタ270は、プロトコル変換機能271により当該メッセージのプロトコルをハブ内部のプロトコルから勘定系システム205のプロトコルに変換し、必要に応じてメッセージ変換機能273によりメッセージ変換を行ない、当該メッセージを勘定系システム205に送信する。勘定系システム205は、受信したメッセージに応じて所定の業務処理を行ない、応答メッセージを作成してアダプタ270に返す。アダプタ270は、当該応答メッセージを受信し、プロトコル変換機能271により当該応答メッセージのプロトコルをハブ内部のプロトコルに変換し、送信先取得機能272によりファインダ250に当該応答メッセージをどこに送信するか送信先を問い合わせる。ここでは送信先はフロー制御AP260である。アダプタ270は、必要に応じてメッセージ変換機能273によりメッセージ変換を行ない、送受信機能274により要求発行元であるフロー制御AP260に応答メッセージを送る。

【0028】

投資信託系システム206へのアクセス処理(262)で送信されたメッセージは、投資信託系システム206に対応するアダプタ280で受信される。アダプタ280での処理は、アダプタ270と同様であるので、説明は省略する。

【0029】

以上のように、通常パスでは、クライアント側システムから発行されたメッセージに対し、クライアント側アダプタでプロトコル変換などの必要な変換を施し、フロー制御APに渡し、フロー制御APが所定の流れでサーバ側アダプタ経由でサーバ側システムにアクセスして業務処理を進めていく。クライアント側およびサーバ側のアダプタとフロー制御APとの間のメッセージの送受信は、ハブ内のプロトコル(CORBA仕様)で行なわれる。

【0030】

次に、アダプタ直結パスについて説明する。上記通常パスでは、フロー制御APで複数のサーバを連携させた処理を行なうことができる。しかし、フロー制御を必要としない場合は、フロー制御APを介することなく、アダプタ直結パスに

より、クライアント側アダプタから直接サーバ側アダプタにメッセージを送ることができる。以下では、営業店システム 2 0 3 から勘定系システム 2 0 5 にメッセージを送る場合を例にしてアダプタ直結パスについて説明する。営業店システム 2 0 3 で用いているプロトコルと勘定系システム 2 0 5 で用いているプロトコルは異なるものとする。

【 0 0 3 1 】

アダプタ 2 2 0 は、営業店システム 2 0 3 に対応したアダプタである。営業店システム 2 0 3 から送信されたメッセージは、アダプタ 2 2 0 で受信される。アダプタ 2 2 0 は、営業店システム 2 0 3 からのメッセージを受信すると、3つのパスのうちどのパスを利用するかを当該メッセージに応じて決定する。ここでは、アダプタ直結パスを利用するメッセージであったとする。その場合、アダプタ 2 2 0 は、プロトコル変換機能 2 2 1 により当該メッセージのプロトコルをハブ内部のプロトコルに変換し、送信先取得（宛先問い合わせ）機能 2 2 2 によりファイнда 2 5 0 にそのメッセージをどこに送信するか送信先を問い合わせる。アダプタ直結パスの場合、ファイнда 2 5 0 は、メッセージを直接送るべきサーバ側システムのアダプタを当該メッセージの送信先として返す。ここでは、送信先として勘定系システム 2 0 5 のアダプタ 2 7 0 が指示されたとする。アダプタ 2 2 0 は、それを受けて、必要に応じてメッセージ変換機能 2 2 3 によりメッセージ変換を行ない、ハブ内メッセージ送受信機能 2 2 4 により当該メッセージを勘定系システム 2 0 5 に対応するアダプタ 2 7 0 に送る。アダプタ 2 7 0 の処理は、通常パスで説明したのと同様である。ただし、応答メッセージは、アダプタ 2 7 0 からアダプタ 2 2 0 に直接返される。図 2 の矢印 2 9 1, 2 9 2 は、アダプタ直結パスでのメッセージの直接のやり取りを示す。

【 0 0 3 2 】

以上のように、アダプタ直結パスでは、クライアント側アダプタから発行されたメッセージが直接サーバ側アダプタに送信され、フロー制御 A P 2 6 0 を経由しない。したがって、通常パスより通信速度が速くレスポンスも速い。なお、上述のアダプタ直結パスでは、クライアント側アダプタ 2 2 0 でクライアント側システム 2 0 3 のプロトコルからハブ内のプロトコル（C O R B A 仕様）に変換し

、サーバ側アダプタ 2 7 0 でハブ内のプロトコルからサーバ側システム 2 0 5 のプロトコルに変換している。そのようなプロトコル変換を行なう代わりに、クライアント側アダプタ 2 2 0 でクライアント側システム 2 0 3 のプロトコルからサーバ側システム 2 0 5 のプロトコルに変換して、ハブ内のプロトコルを介することなくメッセージを直接送ってもよい。これにより、ハブ内のプロトコルを介さない分だけ高速な通信が行なえる。

【 0 0 3 3 】

次に、特定プロトコル直結パスについて説明する。上記アダプタ直結パスでは、クライアント側のプロトコルとサーバ側のプロトコルが異なるため、プロトコル変換が必須であった。しかし、クライアント側のプロトコルとサーバ側のプロトコルが同じ場合は、プロトコル変換を行なうことなく、特定プロトコル直結パスにより、クライアント側アダプタから直接サーバ側アダプタにメッセージを送ることができる。以下では、営業店システム 2 0 3 から勘定系システム 2 0 5 にメッセージを送る場合を例にして特定プロトコル直結パスについて説明する。営業店システム 2 0 3 で用いているプロトコルと勘定系システム 2 0 5 で用いているプロトコルは同じであるものとする。

【 0 0 3 4 】

営業店システム 2 0 3 から送信されたメッセージは、アダプタ 2 2 0 で受信される。アダプタ 2 2 0 は、営業店システム 2 0 3 からのメッセージを受信すると、3 つのパスのうちどのパスを利用するかを当該メッセージに応じて決定する。ここでは、特定プロトコル直結パスを利用するメッセージであったとする。その場合、アダプタ 2 2 0 は直接アダプタ 2 7 0 に当該メッセージを送信し、アダプタ 2 7 0 で当該メッセージを受信する。プロトコル変換は不要であり、アダプタ 2 7 0 は、受信したメッセージを勘定系システム 2 0 5 に送信する。応答メッセージも、同様にして特定プロトコル直結パスで返される。

【 0 0 3 5 】

以上のように、特定プロトコル直結パスでは、クライアント側アダプタから発行されたメッセージが直接サーバ側アダプタに送信され、プロトコル変換の必要もない。すなわち、CORBA でメッセージを交換するハブ内メッセージ送受信

機能 2 2 4, 2 7 4 を用いて通信するのではなく、営業店システム 2 0 3 および勘定系システム 2 0 5 で用いているプロトコルで直接通信する。したがって、アダプタ直結パスより通信速度が速くレスポンスも速い。なお、メッセージ変換は必要に応じてクライアント側アダプタ 2 2 0 またはサーバ側アダプタ 2 7 0 の何れかで行なえばよい。

【 0 0 3 6 】

図 3 は、図 1 および図 2 で説明した本実施の形態におけるハブのシステム構成例を示す。ここでは、3 台の計算機（ハブ・サーバ）3 0 1 ～ 3 0 3 でハブを構成している。ハブ・サーバ 3 0 1 ～ 3 0 3 には、それぞれ、OS（オペレーティング・システム）3 1 1, 3 2 1, 3 3 1 と CORBA（分散オブジェクト通信ミドルウェア）3 1 2, 3 2 2, 3 3 2 が実装されている。ハブ・サーバ 3 0 1 では、クライアント・プログラム 3 1 3、クライアント側アダプタ・プログラム 3 1 4、サーバ側アダプタ・プログラム 3 1 5、およびサーバ・プログラム 3 1 6 が動作している。ハブ・サーバ 3 0 2 では、ファインダ・プログラム 3 2 3、フロー制御プログラム 3 2 4、サーバ側アダプタ・プログラム 3 2 5、およびサーバ・プログラム 3 2 6 が動作している。ハブ・サーバ 3 0 3 では、共通サービス・プログラム 3 3 3、クライアント側アダプタ・プログラム 3 3 4、クライアント・プログラム 3 3 5、およびサーバ・プログラム 3 3 6 が動作している。

【 0 0 3 7 】

クライアント・プログラム 3 1 3, 3 3 5 は、図 1 や図 2 で説明したクライアント側システム（1 1 1 ～ 1 1 3, 2 0 3, 2 0 4）を実現するプログラムである。クライアント側アダプタ・プログラム 3 1 4, 3 3 4 は、図 1 や図 2 で説明したクライアント側アダプタ（1 0 1, 2 2 0, 2 3 0）を実現するプログラムである。サーバ側アダプタ・プログラム 3 1 5, 3 2 5 は、図 1 や図 2 で説明したサーバ側アダプタ（1 0 4, 2 7 0, 2 8 0）を実現するプログラムである。サーバ・プログラム 3 1 6, 3 2 6, 3 3 6 は、図 1 や図 2 で説明したサーバ側システム（1 2 1 ～ 1 2 3, 2 0 5, 2 0 6）を実現するプログラムである。ファインダ・プログラム 3 2 3 は、図 1 や図 2 で説明したファインダ（1 0 2, 2 5 0）を実現するプログラムである。フロー制御プログラム 3 2 4 は、図 1 や図

2で説明したフロー制御AP（103，260）を実現するプログラムである。共通サービス・プログラム333は、図2で説明した共通サービス（240）を実現するプログラムである。

【0038】

これらの各プログラムは、LAN（ローカル・エリア・ネットワーク）で接続された適当な台数の計算機上で動作し、複数の計算機でハブを構成する場合は、各計算機上でどのプログラムを動作させるかは任意である。さらに、各プログラムについてCORBA上で複数の計算機に機能分散してもよい。なお、クライアント・プログラム313，335とサーバ・プログラム316，326，336は、個々の業務処理を行なうプログラムであり、ハブ内に含まれるものではないが、ハブを構成する計算機内にクライアントやサーバを実装することもできるので、図3ではそのような例を示した。各クライアントおよびサーバは、特別なハードウェアを用いた固有の端末などを備えたものでもよい。

【0039】

図4は、本実施の形態におけるアダプタの内部構成を示す。クライアント側アダプタとサーバ側アダプタは、同じ構成とする。図3に示したようにアダプタはハブを構成する任意の計算機上に実装される機能（プログラム）であり、図4ではその計算機の中央処理装置（CPU）401、ディスプレイ402、キーボード403、メモリ404、およびディスク装置407を図示した。

【0040】

メモリ404上のアダプタを構成するプログラムは、アダプタ・プロトコル依存部プログラム（以下、単に依存部と呼ぶ）405とアダプタ・プロトコル非依存部プログラム（以下、単に非依存部と呼ぶ）406に分けられる。依存部405は、メッセージ受付部451、パス判定部452、およびプロトコル変換部453を備える。非依存部406は、宛先問い合わせ（送信先取得）部461、メッセージ変換部462、およびハブ内メッセージ送受信部463を備える。

【0041】

依存部405のメッセージ受付部451は、このアダプタに対応するクライアント側あるいはサーバ側システムのプロトコルで発行されたメッセージの受付処

理を行なう。パス判定部 4 5 2 は、受け付けたメッセージに応じて、パス判定ルール 4 7 1 に基づいて、通常パス、アダプタ直結パス、および特定プロトコル直結パスのうちの何れを利用するかを判定する。プロトコル変換部 4 5 3（図 2 の 2 2 1, 2 3 1, 2 7 1, 2 8 1 に対応）は、クライアント側あるいはサーバ側システムのプロトコルとハブ内の CORBA 仕様のプロトコルとの間の変換を行なう。以上の各部 4 5 1 ~ 4 5 3 は、このアダプタに対応するクライアント側あるいはサーバ側システムのプロトコルに依存した処理が必要な部分である。

【 0 0 4 2 】

非依存部 4 0 6 の宛先問い合わせ部（図 2 の 2 2 2, 2 3 2, 2 7 2, 2 8 2 に対応）4 6 1 は、ファインダにメッセージの送信先を問い合わせる処理を行なう。メッセージ変換部 4 6 2 は、メッセージ変換ルール 4 7 2 に基づいて、クライアント側システムのメッセージ形式とサーバ側システムのメッセージ形式との間のメッセージ形式の変換を行なう。なお、メッセージ変換はクライアントとサーバとの間の任意の位置で行なえばよいから、メッセージ変換部 4 6 2 はクライアント側アダプタまたはサーバ側アダプタの何れか一方に備えれば充分である。また、物理層での通信手順に加えてシステム間のメッセージ形式の変換をも含んで「プロトコル」と言う場合があるが、本実施の形態では「プロトコル」はメッセージ形式の変換を含まないものとする。ハブ内メッセージ送受信部 4 6 3 は、ハブ内での CORBA 仕様のプロトコルにしたがうメッセージの送受信を行なう。以上の各部 4 6 1 ~ 4 6 3 は、このアダプタに対応するクライアント側あるいはサーバ側システムのプロトコルに非依存な処理を行なう部分であり、CORBA 上で動作する部分である。

【 0 0 4 3 】

なお、アダプタを実現する上記各部のプログラムは、別プロセスで実行されるように構成してもよいし、1つのプロセスで実行されるように構成してもよい。

【 0 0 4 4 】

図 5 は、3つのパスの説明図である。

【 0 0 4 5 】

図 5（a）は、特定プロトコル直結パスでのメッセージの流れを示す。上述し

たようにクライアント側システムとサーバ側システムとが同じプロトコルを使用している場合は、特定プロトコル直結パスによりメッセージを送受信する。すなわち、クライアント側システム 5 1 1 から発行されたメッセージは、クライアント側アダプタ 5 1 2 内のメッセージ受付部 5 1 3（図 4 の 4 5 1）で受信され、パス判定部 5 1 4（図 4 の 4 5 2）でパス判定される。この場合、特定プロトコル直結パスと判定され、パス判定部 5 1 4 から直接サーバ側アダプタ 5 1 5 に送信され、アダプタ 5 1 5 からサーバ側システム 5 1 6 に送信される。この一連のメッセージの送受信は、クライアントとサーバのプロトコルで実行され、アダプタ 5 1 2, 5 1 5 では依存部（図 4 の 4 0 5）内のプログラムのみ使用する。依存部内のプロトコル変換部や、非依存部内の宛先問い合わせ部、メッセージ変換部、およびハブ内メッセージ送受信部は使用しない。したがって、非常に高速にメッセージの送受信を行なうことができる。なお、メッセージ変換が必要な場合は、アダプタ 5 1 2 または 5 1 5 の何れかで行なえばよい。

【0 0 4 6】

図 5（b）は、アダプタ直結パスでのメッセージの流れを示す。上述したようにクライアント側システムとサーバ側システムとが異なるプロトコルを使用しているが、フロー制御は必要でない場合は、アダプタ直結パスによりメッセージを送受信する。すなわち、クライアント側システム 5 2 1 から発行されたメッセージは、クライアント側アダプタ 5 2 2 内のメッセージ受付部 5 2 3（図 4 の 4 5 1）で受信され、パス判定部 5 2 4（図 4 の 4 5 2）でパス判定される。この場合、アダプタ直結パスと判定されるので、プロトコル変換部 5 2 5（図 4 の 4 5 3）でハブ内のプロトコルに変換し、宛先問い合わせ部 5 2 6（図 4 の 4 6 1）で送信先がアダプタ 5 2 8 であることを知り、必要に応じてメッセージ変換部 5 2 7（図 4 の 4 6 2）でメッセージ変換して、サーバ側アダプタ 5 2 8 に送る。アダプタ 5 2 8 では、メッセージに対しプロトコル変換などの必要な処理を施してサーバ側システム 5 2 9 に送信する。アダプタ 5 2 2 と 5 2 8 の間にはフロー制御処理が介在しないので、通常パスよりも高速にメッセージの送受信を行なうことができる。

【0 0 4 7】

図 5 (c) は、通常パスでのメッセージの流れを示す。通常パスでは、クライアント側アダプタ 5 3 2 内のメッセージの流れは、図 5 (b) のアダプタ直結パスのアダプタ 5 2 2 と同様である。ただし、パス判定部 5 3 4 では通常パスと判定され、宛先問い合わせ部 5 3 6 で取得する送信先はフロー制御 A P 5 3 8 になる。フロー制御 A P 5 3 8 は、メッセージに応じて、アダプタ 5 3 9 を介してサーバ 5 4 0 にアクセスし、その後、アダプタ 5 4 0 を介してサーバ 5 4 2 にアクセスする、というように複数のサーバを連携させた処理を行なう。

【 0 0 4 8 】

図 6 は、クライアント側アダプタの処理フローを示す。ステップ 6 0 1 でクライアント側システムからメッセージを受け付けると、ステップ 6 0 2 でそのメッセージの種別を取得する。ステップ 6 0 3 では、パス判定情報を取得する。これは、図 4 のパス判定ルール 4 7 1 を参照するということである。次に、ステップ 6 0 4 で、特定プロトコル直結パス用メッセージであるか否か判定する。特定プロトコル直結パス用メッセージでないときは、ステップ 6 0 5 でハブ内で使用している C O R B A 仕様のプロトコルに変換し、ステップ 6 0 6 でファインダに送信先を問い合わせる。ファインダから送信先を取得したら、ステップ 6 0 7 で当該メッセージをその送信先に送信して、処理終了する。なお、クライアント側アダプタでは、送信先がアダプタ直結パスのサーバ側アダプタであるか通常パスのフロー制御 A P であるかを意識することなく、どちらでも処理は変わらない。ファインダから取得した送信先に送信する処理を行なうだけである。ステップ 6 0 4 で特定プロトコル直結パス用メッセージであると判定されたときは、ステップ 6 0 8 で、特定プロトコル直結パスでメッセージ送信し、処理を終了する。

【 0 0 4 9 】

図 7 は、特定プロトコル直結パス経由でメッセージを受信するサーバ側アダプタの処理フローを示す。ステップ 7 0 1 でメッセージを受信する。これは、クライアントやサーバ固有のプロトコルである特定プロトコルに依存する処理を行なう依存部（図 4 の 4 0 5）でメッセージを受信するものである。ステップ 7 0 2 で、依存部は、特定プロトコルで当該メッセージをサーバ側システムに送信する。サーバ側システムから業務処理の結果の応答メッセージが返答されたら、ステ

ップ 7 0 3 でその応答メッセージを受信する。ステップ 7 0 4 で、依存部は、当該応答メッセージを呼び出し元のクライアント側アダプタに返し、処理を終了する。なお、図 7 ではメッセージ変換を行なわない場合を示したが、メッセージ変換が必要なときは図 7 の処理中で行なうようにしてもよい。

【 0 0 5 0 】

図 8 は、アダプタ直結パスまたは通常パス経由でメッセージを受信するサーバ側アダプタの処理フローを示す。ステップ 8 0 1 でメッセージを受信する。これは、ハブ内で使用されている CORBA 仕様のプロトコルでメッセージを受信する処理であり、図 4 の非依存部 4 0 6 のハブ内メッセージ送受信部 4 6 3 によるメッセージの受信である。次にステップ 8 0 2 で、メッセージ変換が必要か否か判別する。必要な場合は、ステップ 8 0 3 でメッセージ変換を行なう。次に、ステップ 8 0 4 で、メッセージを非依存部から依存部（図 4 の 4 0 5）に渡す。ステップ 8 0 5 で、依存部は、サーバに固有のプロトコルに変換して当該メッセージをサーバ側システムに送信する。サーバ側システムから業務処理の結果の応答メッセージが返答されたら、ステップ 8 0 6 でその応答メッセージを受信する。ステップ 8 0 7 でその応答メッセージをハブ内の CORBA 仕様のプロトコルに変換し、ステップ 8 0 8 で当該応答メッセージを呼び出し元のクライアント側アダプタに返し、処理を終了する。なお、応答メッセージについても、必要に応じてメッセージ変換を行なってもよい。

【 0 0 5 1 】

図 9 は、本実施の形態で授受されるメッセージの例を示す。メッセージは、制御情報 9 0 1、業務コード 9 0 2、および業務固有情報 9 0 3 を備えている。制御情報 9 0 1 は、当該メッセージの送り元および送り先の情報、あるいはデータ長や形式などを表す制御情報である。業務コード 9 0 2 は、どのような業務を依頼するメッセージであるかを示すコード情報である。業務固有情報 9 0 3 は、依頼する業務に固有の各種の情報である。

【 0 0 5 2 】

上記図 6 のクライアント側アダプタの処理フローにおいて、ステップ 6 0 4 のパス判定は、例えば図 9 の制御情報 9 0 1 や業務コード 9 0 2 に基づいて行なう

。①送り元と送り先とが同一プロトコルのシステムで、かつ、業務コードが「普通預金預け入れ」や「普通預金引き出し」であったとき、特定プロトコル直結パスを用いる、②送り元と送り先とが異なるプロトコルのシステムで、かつ、業務コードが「普通預金預け入れ」や「普通預金引き出し」であったとき、アダプタ直結直結パスを用いる、③業務コードが「投資信託申し込み」であったとき（勘定系システムと投資信託系システムの両方にアクセスが必要）、通常パスを用いる、というような具合にパス判定する。さらに、例えば業務コードが「普通預金預け入れ」のときは業務固有情報としてその預け入れ金額が設定されているので、その金額に応じてパス判定するようなことが行なえる。金額が所定値より少ないときは特定プロトコル直結パスを用い、金額が所定値以上のときは通常パスを用いてクライアント側に広告を出したり顧客分析システムを連動させるシステムにもアクセスする、というようなことが行なえる。そのようなパス判定のルールは、図 4 のパス判定ルール 4 7 1 に設定しておく。

【 0 0 5 3 】

なお、通信制御手段で C O R B A 仕様を用いているが、これはアダプタ間の通信を制御できるものであれば C O R B A には限定しない。

【 0 0 5 4 】

【発明の効果】

以上説明したように、本発明によれば、受け付けたメッセージに応じてパス判定し、特定プロトコル直結パスが選択されたときには、プロトコル変換を施すことなく直接メッセージを送信するので、連携システムを経由するメッセージの通信時間や応答メッセージを得るまでのレスポンス時間を短くすることができる。

【図面の簡単な説明】

【図 1】

本発明の実施の形態であるハブ・アンド・スポーク・システムの概要図

【図 2】

ハブ・アンド・スポーク・システムの実行環境の概要図

【図 3】

ハブのシステム構成例を示す図

【図 4】

アダプタの内部構成を示す図

【図 5】

3 つのパスの説明図

【図 6】

クライアント側アダプタの処理フロー図

【図 7】

特定プロトコル直結パス経由でメッセージを受信するサーバ側アダプタの処理フロー図

【図 8】

アダプタ直結パスまたは通常パス経由でメッセージを受信するサーバ側アダプタの処理フロー図

【図 9】

本実施の形態で授受されるメッセージの例を示す図

【図 1 0】

従来の情報システム間の連携の一例を示す図

【図 1 1】

従来のハブ・アンド・スポークでの接続例を示す図

【符号の説明】

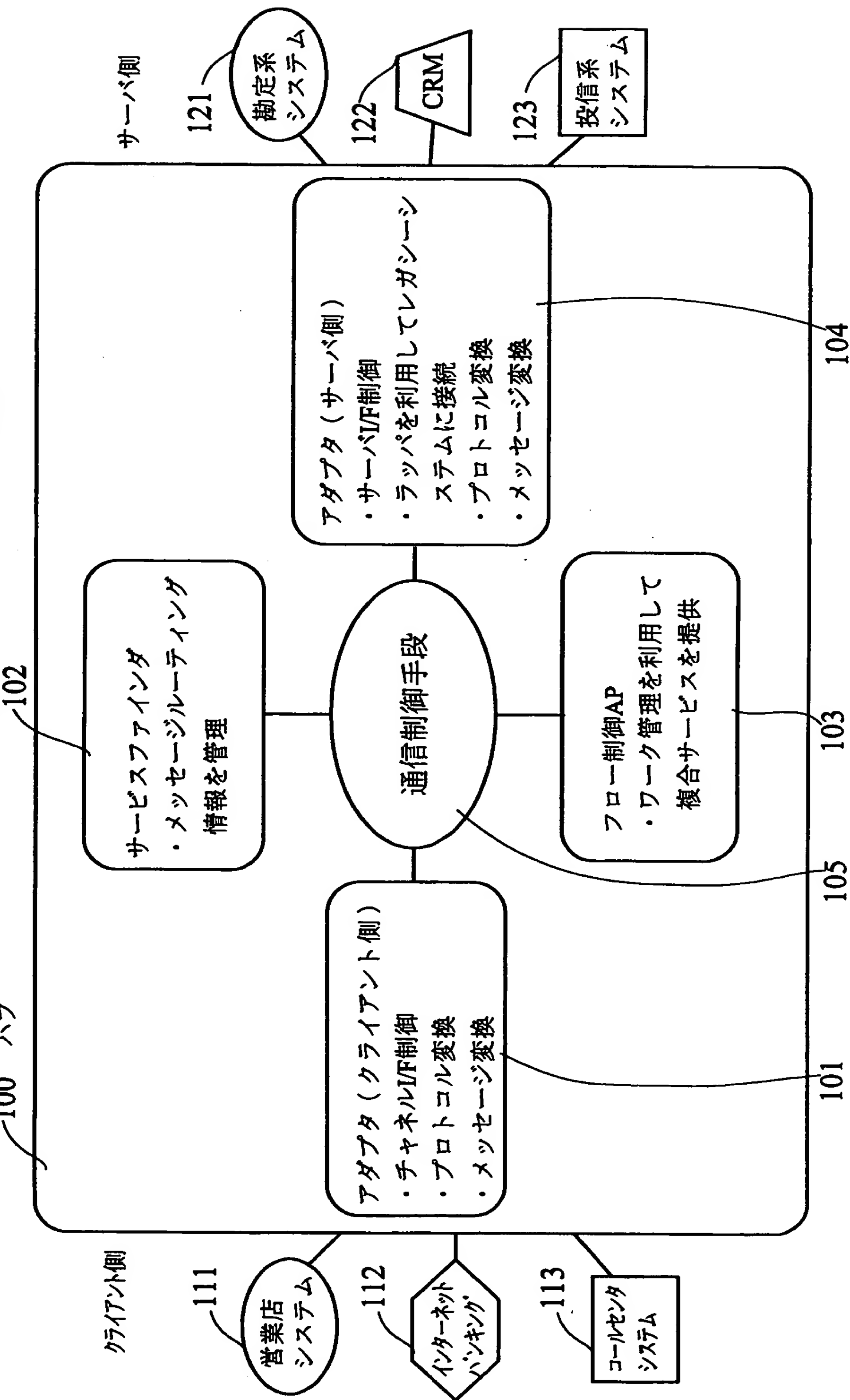
1 0 0…ハブ、1 1 1…営業店システム、1 1 2…インターネット・バンキング・システム、1 1 3…コール・センタ・システム、1 2 1…勘定系システム、1 2 2…CRMシステム、1 2 3…投資信託系システム、1 0 1…クライアント側アダプタ、1 0 2…サービス・ファインダ、1 0 3…フロー制御AP、1 0 4…サーバ側アダプタ、1 0 5…CORBA。

特平 1 1 - 1 2 3 5 0 8

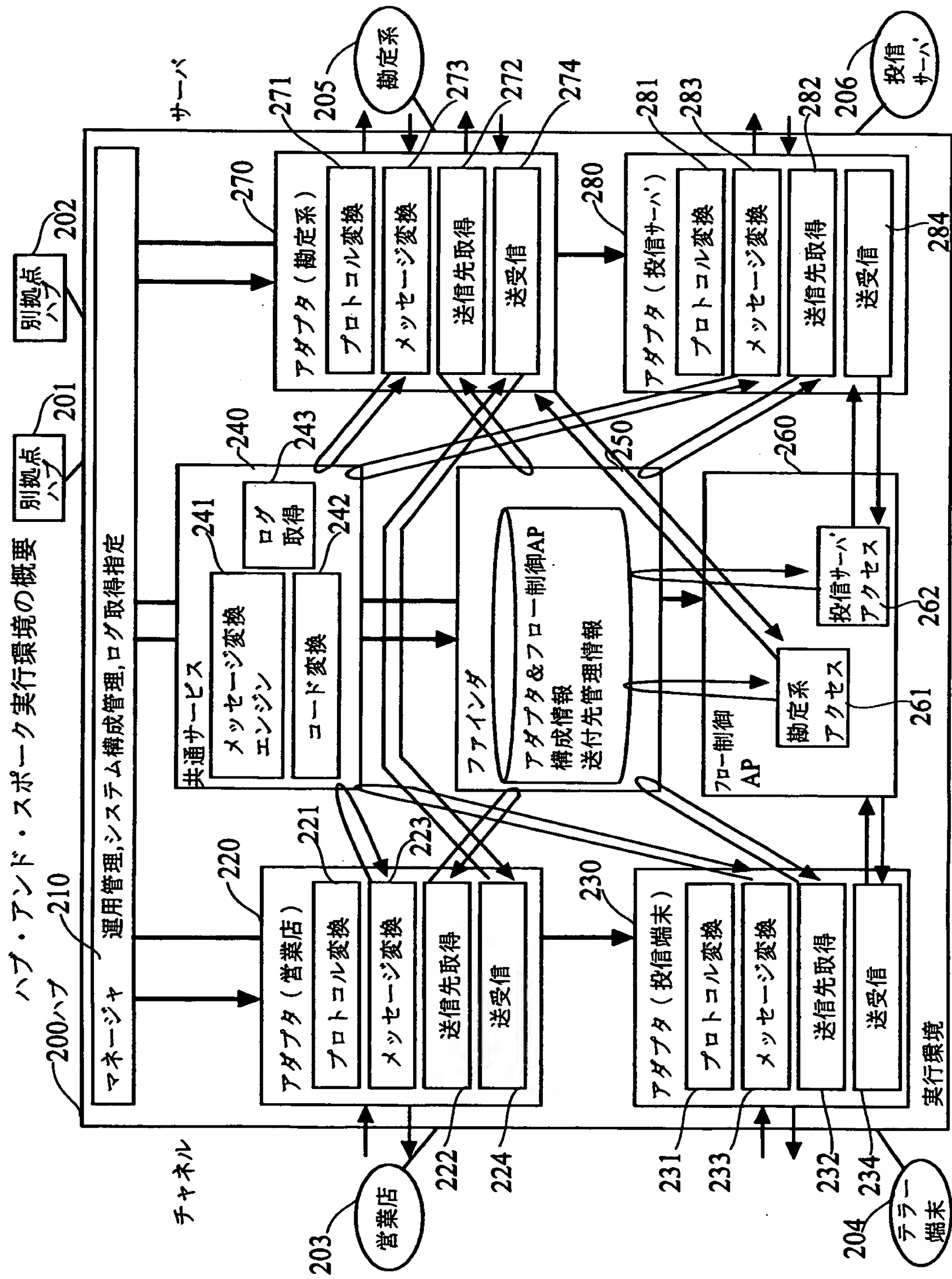
【書類名】 図面

【図 1】

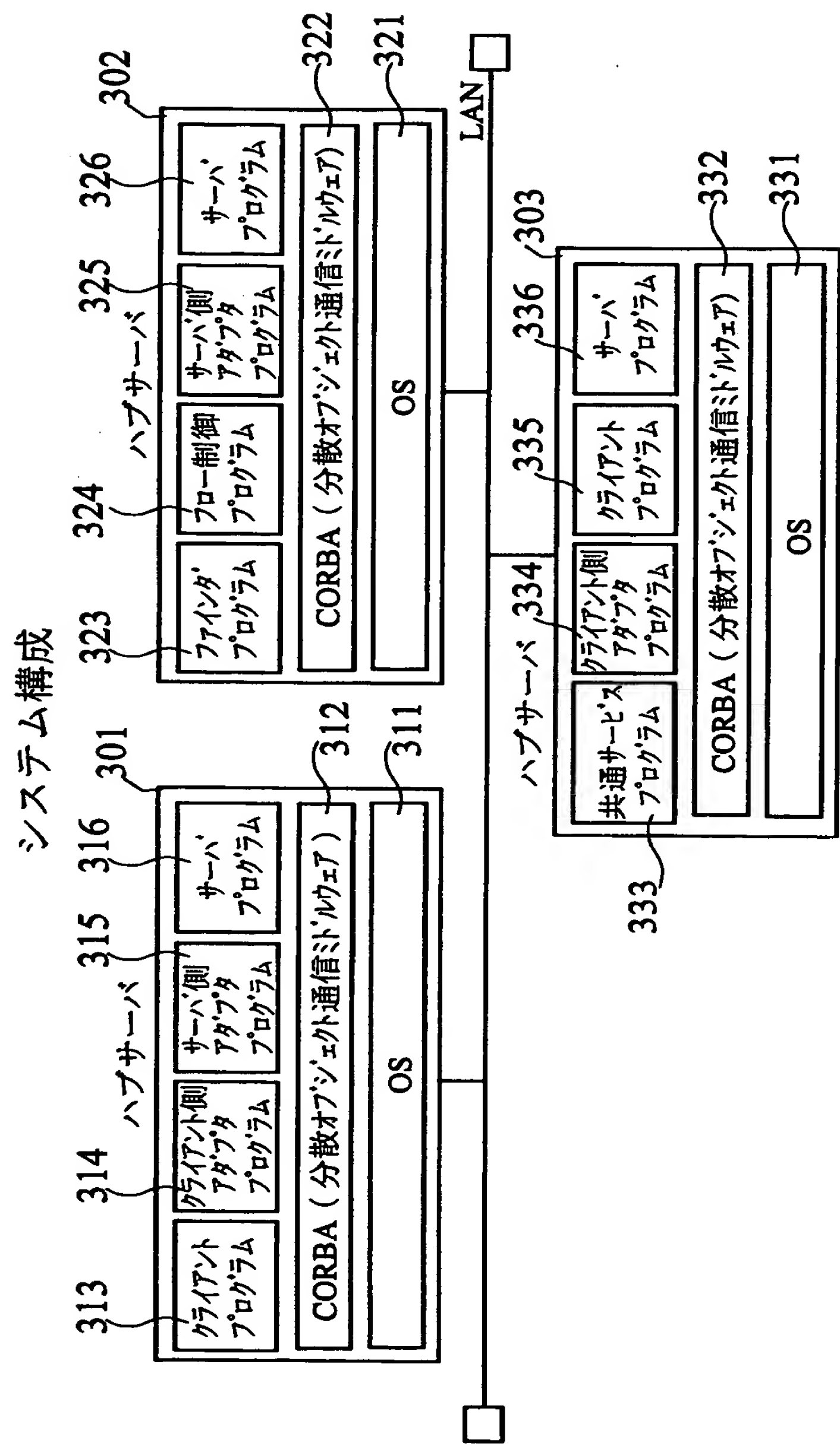
100 ハブ ハブ・アンド・スポーク・システムの概要



【図 2】

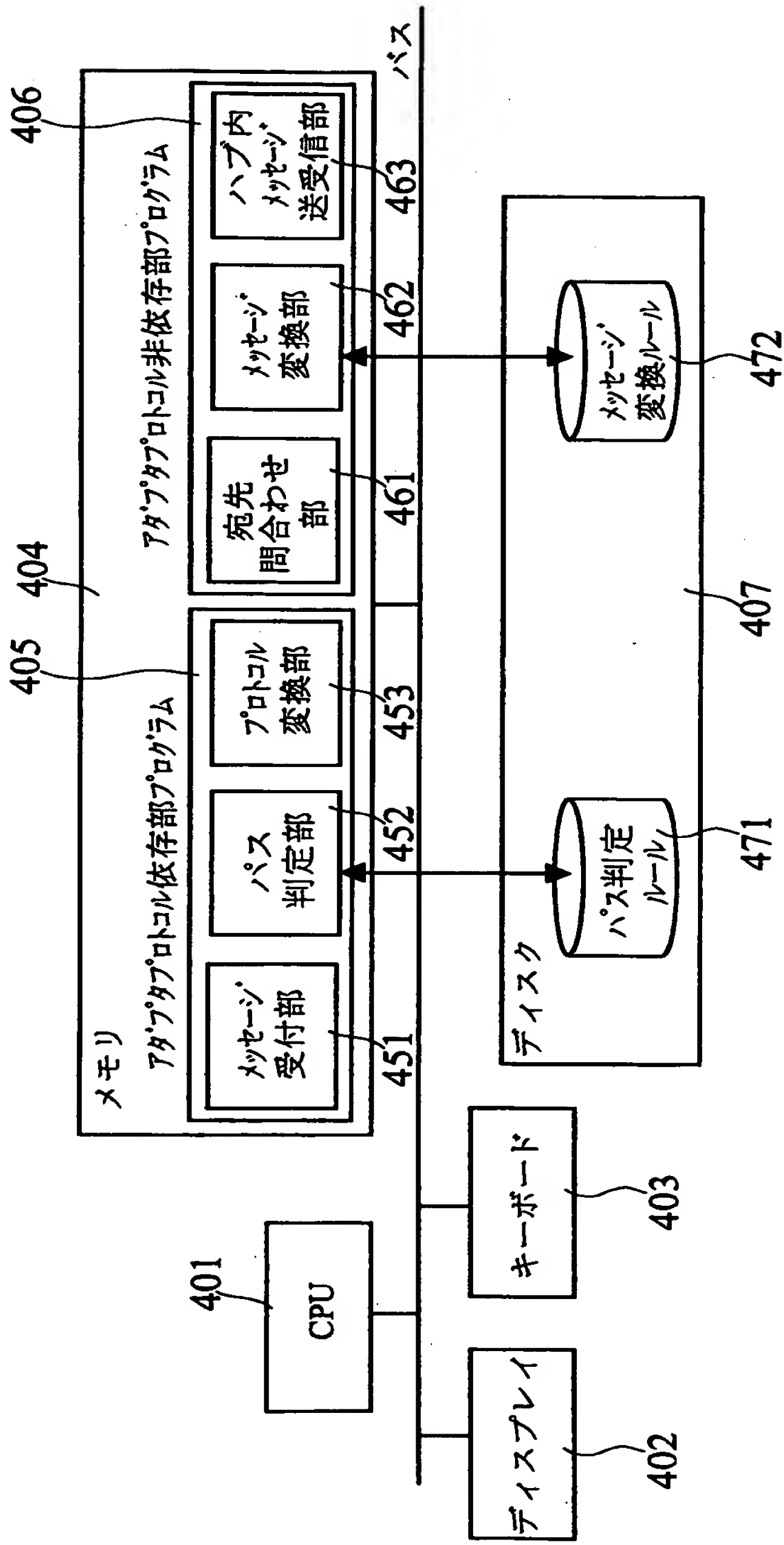


【図 3】



【図 4】

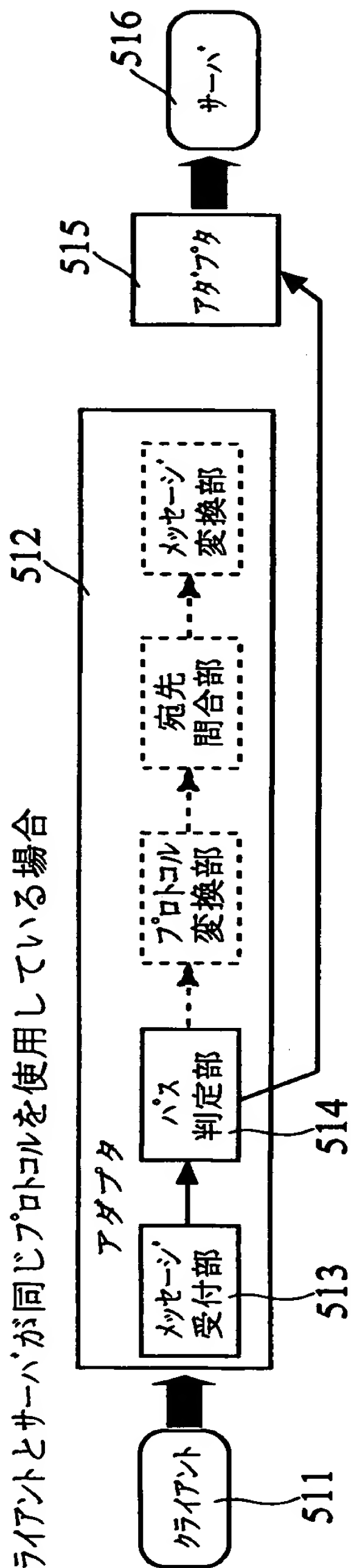
アダプタの内部構成



【図 5】

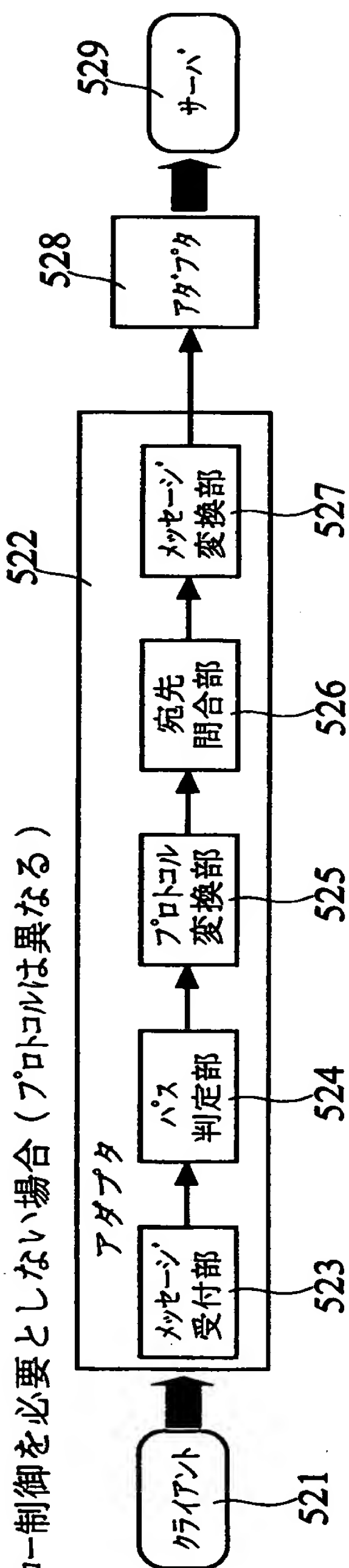
(a) 特定プロトコル直結パス

クライアントとサーバが同じプロトコルを使用している場合

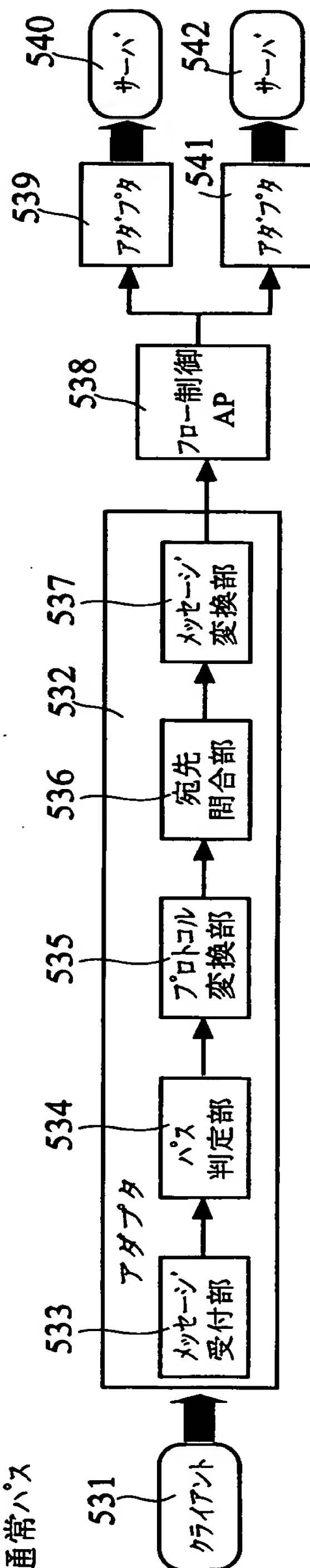


(b) アダプタ直結パス

フロー制御を必要としない場合 (プロトコルは異なる)

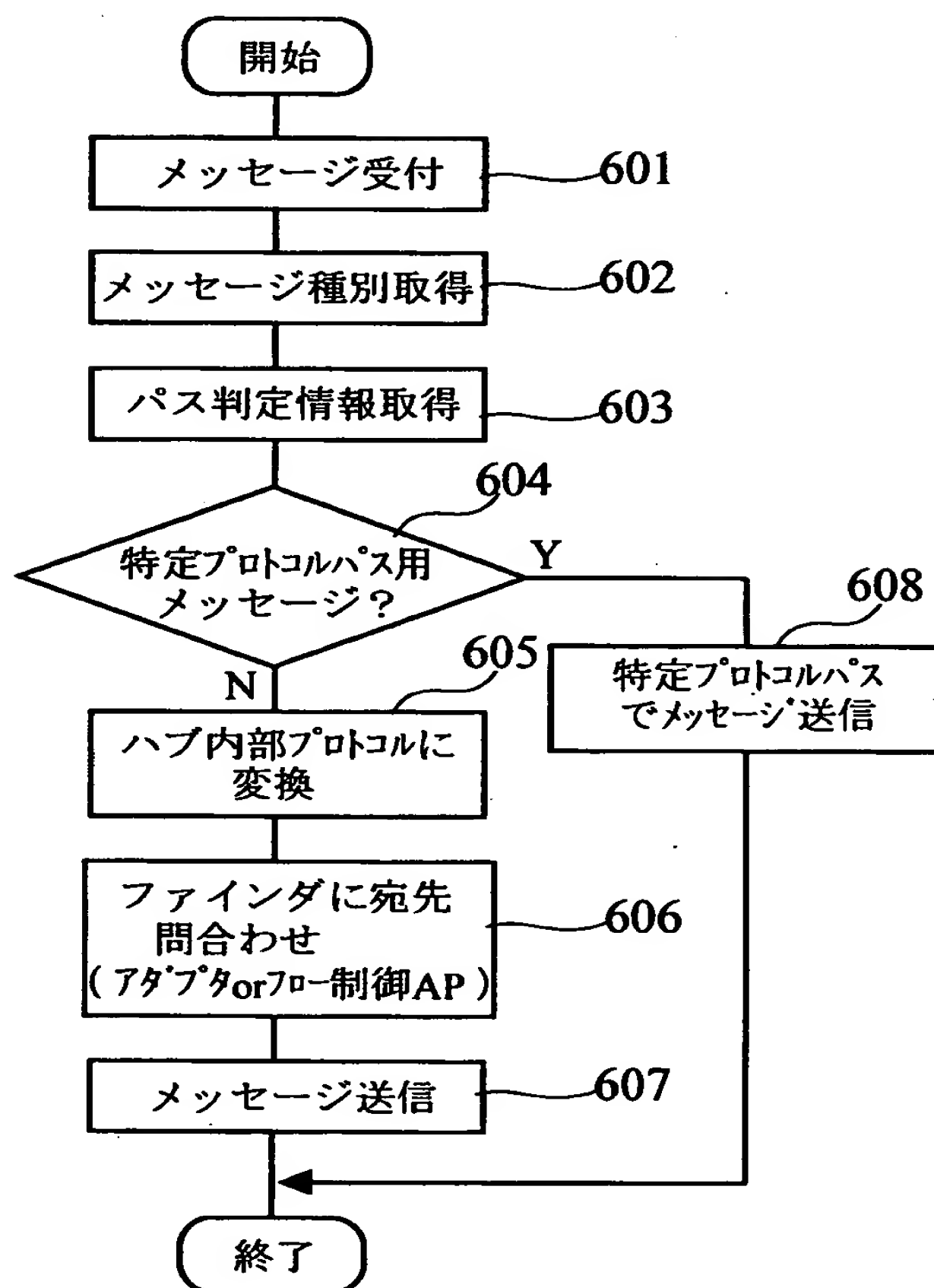


(c) 通常パス



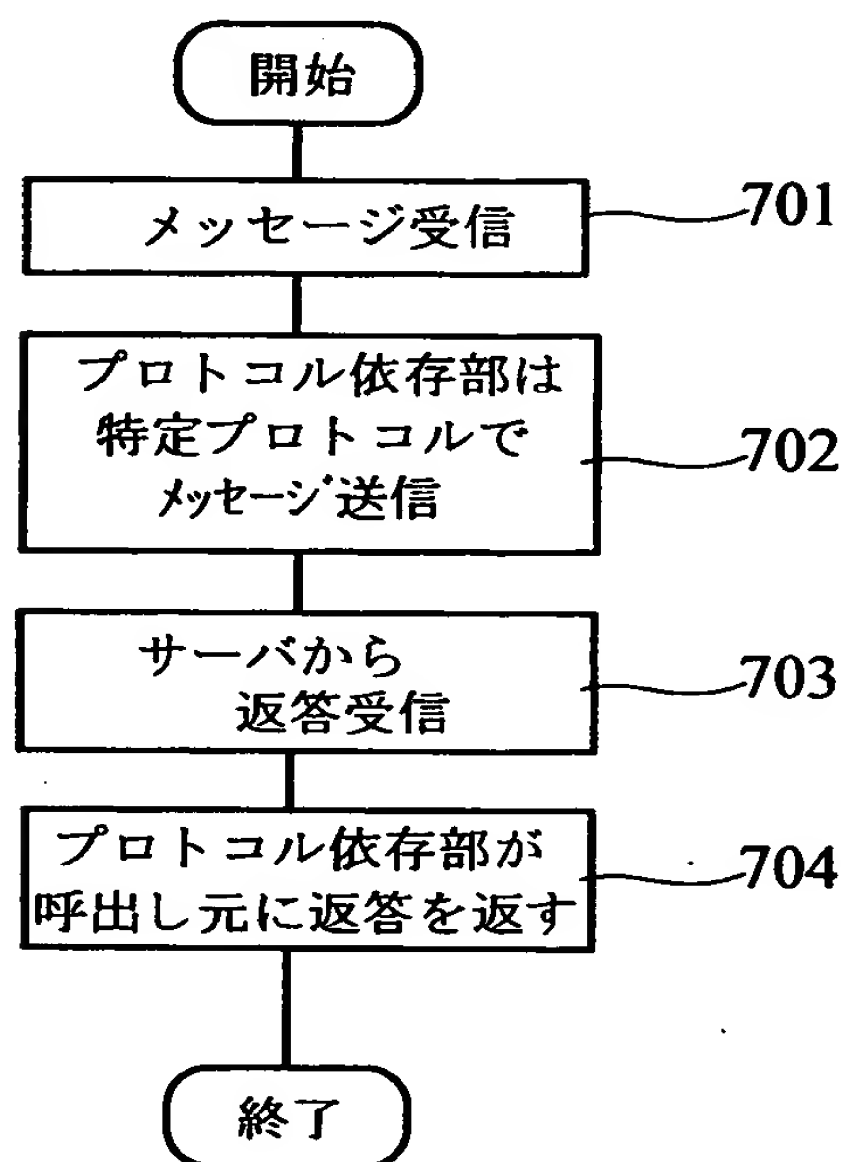
【図 6】

アダプタの処理フロー（クライアント側）



【図 7】

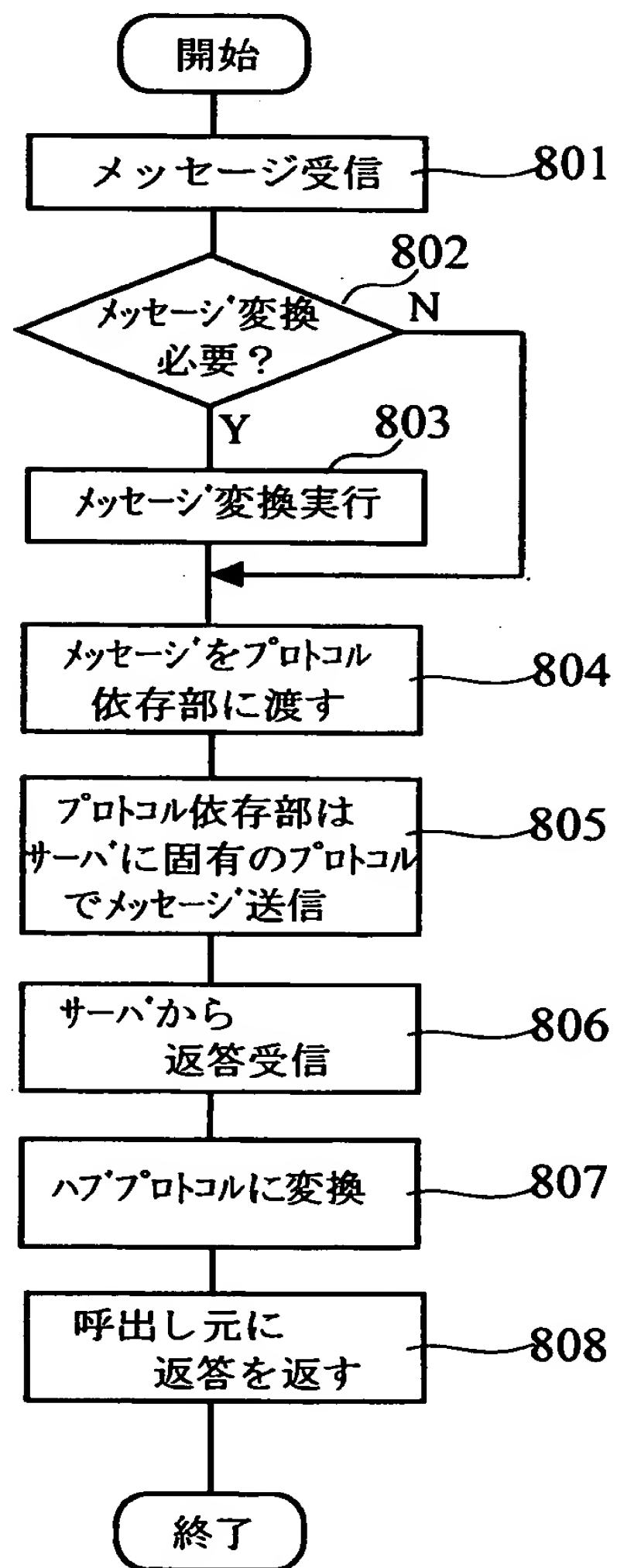
アダプタの処理フロー（サーバ側）
特定プロトコル直結パスの場合



【図 8】

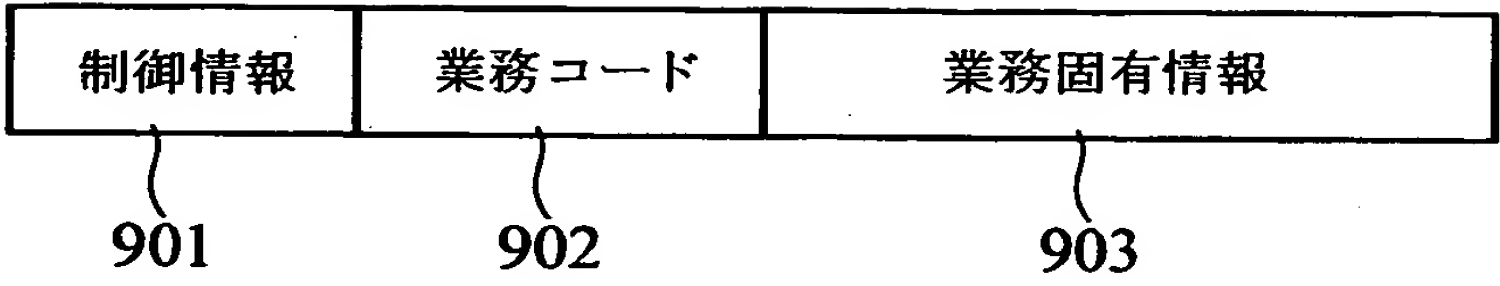
アダプタの処理フロー（サーバ側）

アダプタ直結パスまたは通常パスの場合



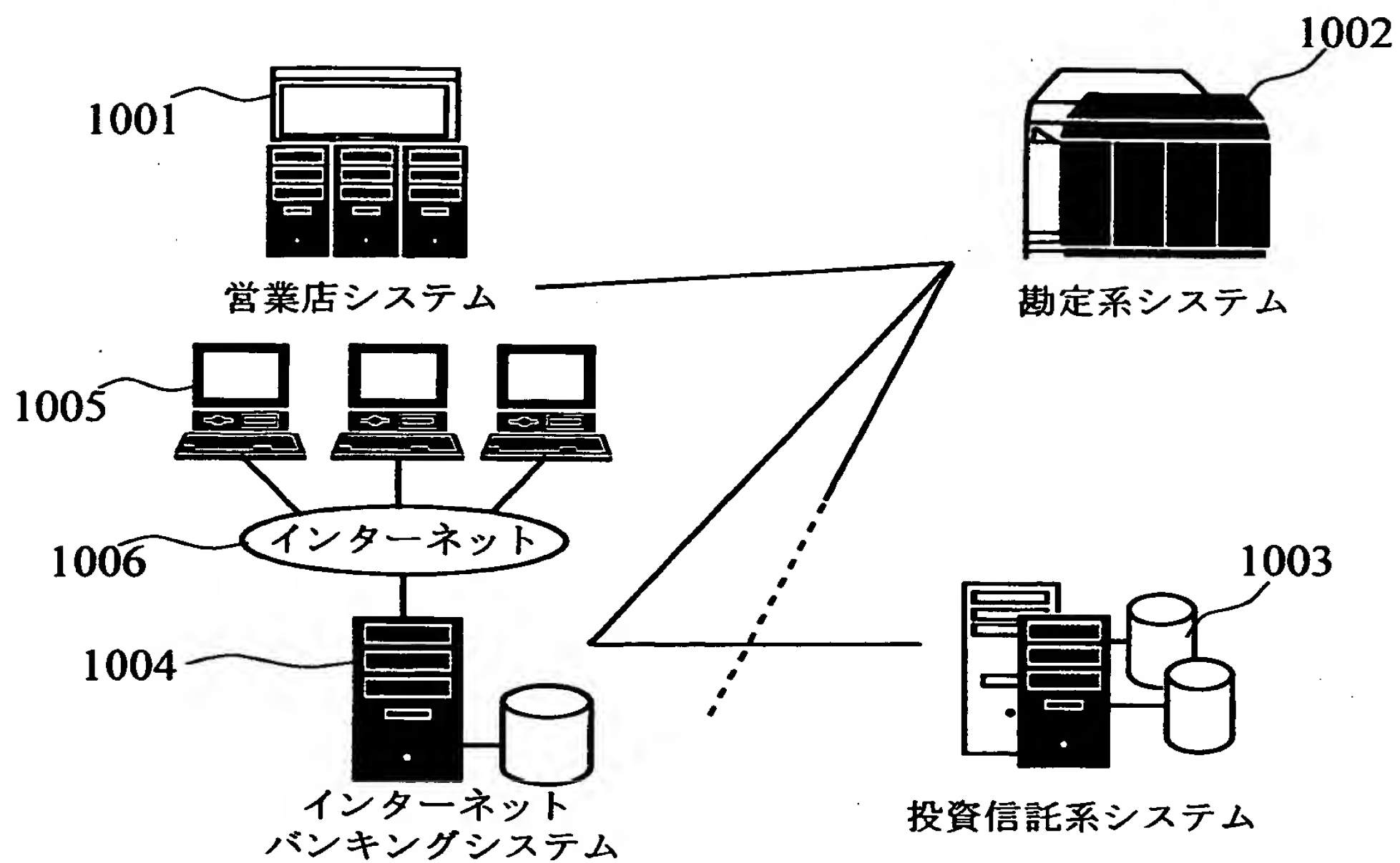
【図 9】

メッセージ例



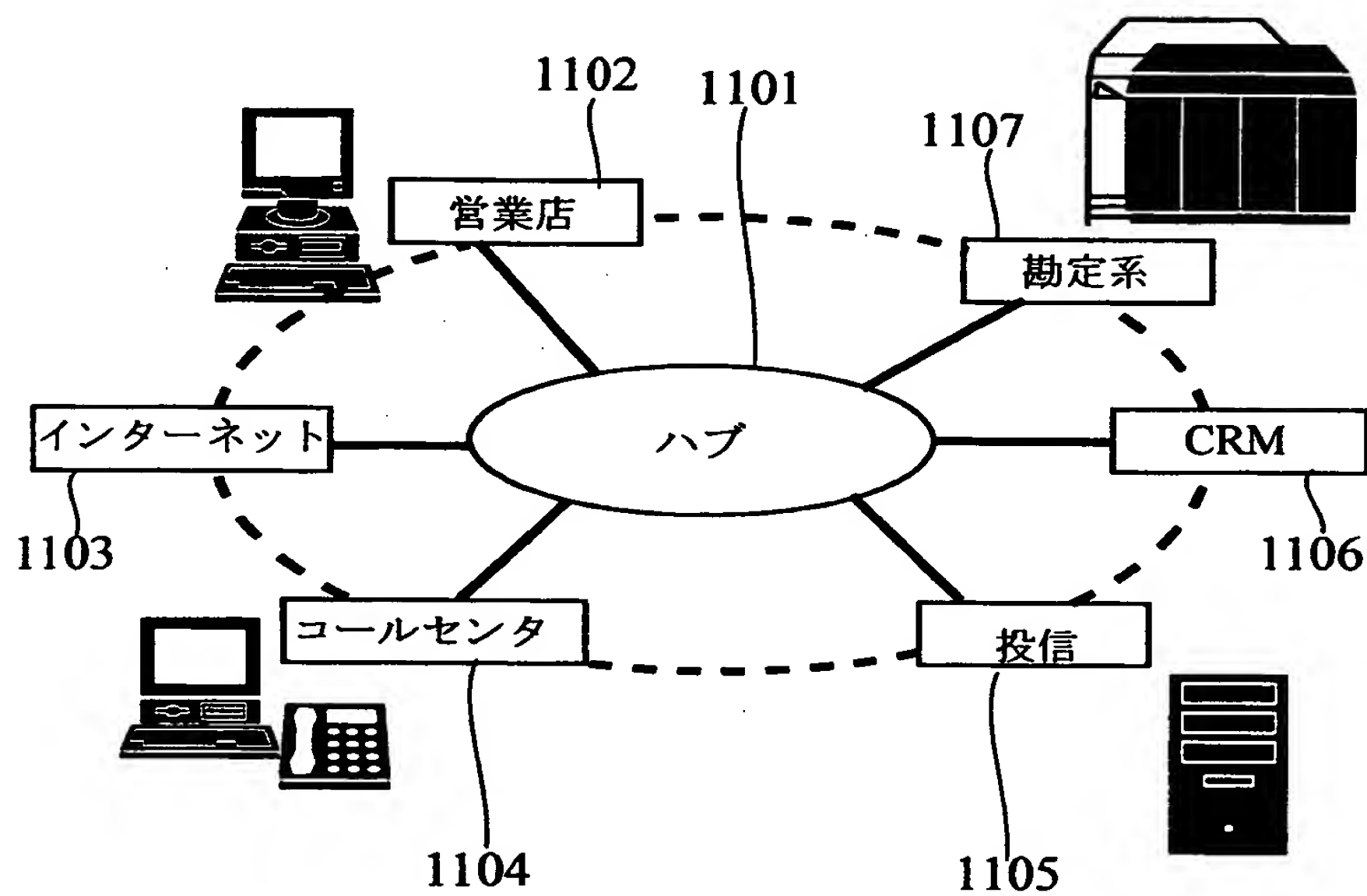
【図 1 0】

従来の情報システム間の連携の一例



【図 1 1】

ハブ・アンド・スポーク方式での接続例



【書類名】 要約書

【要約】

【課題】

従来より、各種の情報システムを経路制御やメッセージの変換を行なう機能をもつコアとなるシステムに接続し、このコアとなるシステムを介してシステム間を連携させる方式が提案されている。ここでは、このような方式をハブ・アンド・スポークと呼び、コアの部分をハブと呼ぶものとする。ハブ・アンド・スポーク方式のシステムでは各システム間の差違をハブが吸収するので、各システムはハブに接続すれば連携が容易になるが、従来のハブでは、どのようなメッセージが入力した場合でも、ハブ内で経路判定しプロトコル変換およびメッセージ変換を施して相手先のシステムに転送するため、メッセージ通信の時間や応答メッセージを得るまでのレスポンス時間が長くなるという問題があった。

【解決手段】

任意の数の第 1 の情報システムと任意の数の第 2 の情報システムとの間に介在して、第 1 の情報システムと第 2 の情報システムとを連携させて動作させるシステム間連携システムにおいて、第 1 の情報システムから送信されるメッセージに対し、第 2 の情報システムに渡す経路として、特定プロトコル直結パスまたはそれ以外のパスの何れかを、受け付けたメッセージに基づいて選択する。第 1 の情報システムと第 2 の情報システムとが同じプロトコルを使用するものである場合は、特定プロトコル直結パスを選択し、プロトコル変換を施すことなく直接メッセージを転送する。そうでないときは、特定プロトコル直結パス以外のパスを選択し、いったん連携システム内のプロトコルに変換して転送する。

【選択図】 図 2

認定・付加情報

特許出願の番号	平成 1 1 年	特許願	第 1 2 3 5 0 8 号
受付番号	5 9 9 0 0 4 1 7 5 3 0		
書類名	特許願		
担当官	第七担当上席	0 0 9 6	
作成日	平成 1 1 年	5 月 1 0 日	

<認定情報・付加情報>

【提出日】	平成11年 4月30日
-------	-------------

特平 1 1 - 1 2 3 5 0 8

出 願 人 履 歴 情 報

識別番号 [0 0 0 0 0 5 1 0 8]

1. 変更年月日	1 9 9 0 年 8 月 3 1 日
[変更理由]	新規登録
住 所	東京都千代田区神田駿河台 4 丁目 6 番地
氏 名	株式会社日立製作所